

**Amendments to the Claims**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing of Claims:**

1. (currently amended) A method for finding ~~an~~ a global extrema for an n-dimensional array having a plurality of processing elements, the method comprising:

determining within each of said processing elements a dimensional extrema for a first dimension of said n-dimensional array, wherein said dimensional extrema is related to one or more local extrema of said processing elements in said first dimension;

determining within each of said processing elements a next dimensional extrema for a next dimension of said n-dimensional array, wherein said next dimensional extrema is related to one or more of said first dimensional extrema; ~~and~~

repeating said determining within each of said processing elements a next dimensional extrema for each of said n-dimensions, wherein each of said next dimensional extrema is related to a dimensional extrema from a previously selected dimension, until the global extrema is found;  
and

saving said global extrema.

2. (currently amended) The method of claim 1 further comprising determining a local extrema for each of said processing ~~element~~ elements.

3. (currently amended) The method of claim 2 wherein said determining a local extrema for at least one of said processing elements comprises:

separating a set of input values within said processing element into an odd ~~numbered~~ set corresponding to values in odd positions within said set of input values and an even ~~numbered~~ set corresponding to values in even positions within said set of input values;

determining ~~an odd~~ a first extrema from said odd ~~numbered~~ set;

determining ~~an even~~ a second extrema from said even ~~numbered~~ set; and

determining said local extrema from said ~~odd~~ first extrema and said ~~even~~ second extrema.

4. (currently amended) The method of claim 1 wherein said determining within each of said processing elements ~~a dimensional extremas~~ extrema for a first dimension of said n-dimensional array comprises:

receiving a set of local extrema from one or more of said processing elements within said first dimension;

separating said set of local extrema into an odd ~~numbered~~ set corresponding to values in odd positions within said set of local extrema and an even ~~numbered~~ set corresponding to values in even positions within said set of local extrema;

determining ~~an odd~~ a first extrema from said odd ~~numbered~~ set;

determining ~~an even~~ a second extrema from said even ~~numbered~~ set; and

determining said dimensional extrema for a first dimension from said ~~odd~~ first extrema and said ~~even~~ second extrema.

5. (currently amended) The method of claim 1 wherein said determining within each of said processing elements a next dimensional extrema for a next dimension of said n-dimensional array comprises:

receiving a set of said dimensional extrema for a first dimension from one or more of said processing elements within said next dimension;

separating said set of dimensional extrema for a first dimension into an odd ~~numbered~~ set corresponding to values in odd positions within said set of dimensional extrema and an even ~~numbered~~ set corresponding to values in even positions within said set of dimensional extrema;

determining ~~a odd~~ a first extrema from said odd ~~numbered~~ set;

determining ~~an even~~ a second extrema from said even ~~numbered~~ set; and

determining said next dimensional extrema for a next dimension from said ~~odd~~ first extrema and said ~~even~~ second extrema.

6. (currently amended) The method of claim 1 wherein said repeating said determining within each of said processing elements a next dimensional extrema for each of n-said dimensions comprises:

receiving a set of dimensional extrema from one or more of said processing elements within a previously selected dimension;

separating said set of dimensional extrema from said previously selected dimension into an odd ~~numbered~~ set corresponding to values in odd positions within said set of dimensional extrema and an even ~~numbered~~ set corresponding to values in even positions within said set of dimensional extrema;

determining ~~a odd~~ a first extrema from said odd ~~numbered~~ set;

determining ~~an even~~ a second extrema from said even ~~numbered~~ set; and

determining said next dimensional extrema for said next dimension from said ~~odd~~ first extrema and said ~~even~~ second extrema.

7. (currently amended) The method of claim [[4]] 1 wherein determining within each of said processing elements a dimensional extrema for a first dimension of said n-dimensional array comprises:

loading a value from an odd position within said set into a first register;

loading a value from an even position within said set into a second register and transferring said value within said first register to a third register;

loading a value from a next odd position within said set into said first register and transferring said value within said second register to a fourth register; and

loading a value from a next even position within said set into said second register.

8. (currently amended) The method of claim [[4]] 7 wherein said determining ~~an odd~~ a first extrema from said odd ~~numbered~~ set comprises comparing the value in said first register to the value in said third register.

9. (currently amended) The method of claim 8 further comprising:

selecting the greater value from said first register and said third register if a high ~~odd~~ first extrema is desired; and

selecting the lesser value from said first register and said third register if a low ~~odd~~ first extrema is desired.

10. (currently amended) The method of claim [[4]] 7 wherein said determining ~~an even~~ a second extrema from said even set of values comprises comparing the value in said second register to the value in said fourth register.

11. (currently amended) The method of claim 10 further comprising:

selecting the greater value from said second register and said fourth register if ~~the a~~ a high ~~even~~ second extrema is desired; and

selecting the lesser value from said second register and said fourth register if ~~the a~~ a low ~~even~~ second extrema is desired.

12. (currently amended) The method of claim 8 further comprising:

updating said ~~odd~~ first extrema in said third register;  
loading another value from an odd position within said set into said first register;  
comparing the value in said first register to the value in said third register; and  
repeating said updating, loading and comparing steps for remaining values within an odd position within said set.

13. (currently amended) The method of claim 10 ~~wherein said~~ further comprising:  
updating said ~~even~~ second extrema in said fourth register;  
loading another value from an even position within said set into said second register;  
comparing the value in said second register to the value in said fourth register; and  
repeating said updating, loading and comparing steps for remaining values within an even position within said set.
14. (currently amended) A method comprising:  
identifying extrema within a data stream as having one of an odd or an even position;  
processing said extrema having an odd position to produce ~~an odd~~ a first extrema;  
processing said extrema having an even position to produce ~~an even~~ a second extrema; ~~and~~  
determining a dimensional extrema from said ~~odd~~ first extrema and said ~~even~~ second extrema; ~~and~~  
storing said dimensional extrema.
15. (currently amended) The method of claim 14 wherein said processing said extrema having an odd position and processing said extrema having an even position comprises:  
loading a extrema having an odd position into a first register;  
loading a extrema having an even position into a second register and transferring said extrema within said first register into a third register;  
loading a extrema from a next odd position within said data stream into said first register and transferring said extrema within said second register into a fourth register;  
comparing said extrema in said first register to said extrema in said third register to produce said ~~odd~~ first extrema and loading a extrema from a next even position within said data stream into said second register; and  
comparing said extrema in said second register to said extrema in said fourth register to produce said ~~even~~ second extrema.

16. (currently amended) The method of claim 15 wherein said processing said extrema having an odd position to produce ~~an odd~~ a first extrema further comprises:
- selecting the greater valued extrema from said first register and said third register if a high ~~odd~~ first extrema is desired; and
  - selecting the lesser valued extrema from said first register and said third register if a low ~~odd~~ first extrema is desired.
17. (currently amended) The method of claim 15 wherein said processing said extrema having an even position to produce ~~an even~~ a second extrema further comprises:
- selecting the greater valued extrema from said second register and said fourth register if a high ~~even~~ second extrema is desired; and
  - selecting the lesser valued extrema from said second register and said fourth register if a low ~~even~~ second extrema is desired.
18. (currently amended) The method of claim 15 wherein said determining a dimensional extrema from said ~~odd first~~ extrema and said ~~even second~~ extrema further comprises:
- selecting the greater valued extrema from said ~~odd first~~ extrema and said ~~even second~~ extrema if a dimensional high extrema is desired; and
  - selecting the lesser valued extrema from said ~~odd first~~ extrema and said ~~even second~~ extrema if the dimensional low extrema is desired.
19. (currently amended) The method of claim 15 further comprising:
- storing said ~~odd first~~ extrema in said third register;
  - loading another extrema from an odd position within said data stream into said first register;
  - comparing the extrema within said first register to the ~~odd~~ extrema within said third register;
  - and
  - repeating said storing, loading and comparing steps for remaining extrema within an odd position within said data stream.
20. (currently amended) The method of claim 15 further comprising:
- storing said ~~even second~~ extrema in said fourth register;

loading another extrema from an even position within said data stream into said second register;

comparing the extrema within said second register to the ~~even~~ extrema within said fourth register; and

repeating said storing, loading and comparing steps for remaining extrema within an even position within said data stream.

21. (original) The method of claim 14 further comprising:

determining a next dimensional extrema for a next dimension for a n-dimensional array, wherein said next dimensional extrema is related to said dimensional extrema; and

repeating said determining a next dimensional extrema for each of said n-dimensions, wherein each of said next dimensional extrema is related to a dimensional extrema from a previously selected dimension.

22. (currently amended) A method for determining a dimensional extrema for an n-dimensional array of processing elements, comprising:

loading ~~odd-numbered~~ extrema from odd positions within a set of said processing elements in a first dimension into a first plurality of registers;

loading ~~even-numbered~~ extrema from even positions within said a set of set processing elements into a second plurality of registers;

comparing ~~certain of said loaded odd-numbered extrema to produce an odd extrema~~ the extrema loaded into said first plurality of registers to produce a first extrema;

comparing ~~certain of said loaded even-numbered extrema to produce an even extrema; and~~ the extrema loaded into said second plurality of registers to produce a second extrema;

producing a dimensional extrema in response to said ~~odd~~ first extrema and said ~~even~~ second extrema; and

storing said dimensional extrema.

23. (currently amended) The method of claim 22 wherein said loading ~~odd-numbered~~ extrema from odd positions within a set of said processing elements in a first dimension into a first plurality of registers comprises:

loading an extrema having an odd position into a first register;

transferring said extrema in said first register into a third register; and

loading an extrema from a next odd position ~~within said data stream~~ into said first register.

24. (currently amended) The method of claim 22 wherein said loading ~~even-numbered~~ extrema from even positions within a set of set processing elements into a second plurality of registers comprises:

loading an extrema having an even position into a second register;

transferring said extrema in said second register into a fourth register; and

loading an extrema from a next even position ~~within said data stream~~ into said second register.

25. (currently amended) The method of claim 23 wherein said comparing ~~certain of said loaded odd numbered extrema~~ to produce an ~~odd~~ said first extrema comprises comparing said extrema in said first register to said extrema in said third register to produce said ~~odd~~ first extrema.

26. (currently amended) The method of claim 24 wherein said comparing ~~certain of said loaded even numbered extrema~~ to produce an ~~even~~ said second extrema comprises comparing said extrema in said second register to said extrema in said fourth register to produce said ~~even~~ second extrema.

27. (currently amended) The method of claim 25 wherein said comparing ~~certain of said loaded odd numbered extrema~~ to produce an ~~odd~~ said first extrema further comprises:

storing said ~~odd~~ first extrema in said third register;

loading another extrema from an odd position ~~within said data stream~~ into said first register;

comparing said extrema within said first register to said ~~local~~ extrema within said third register; and

repeating said storing, loading and comparing steps for remaining extrema within an odd position ~~within said data stream~~.

28. (currently amended) The method of claim 26 wherein said comparing ~~certain of said loaded even numbered extrema~~ to produce an ~~even~~ said second extrema further comprises:

storing said ~~even~~ second extrema in said fourth register;

loading another extrema from an even position ~~within said data stream~~ into said second register;

comparing said extrema within said second register to said ~~local~~ extrema within said fourth register; and

repeating said storing, loading and comparing steps for remaining extrema within an even position ~~within said data stream.~~

29. (currently amended) The method of claim 21 22 further comprising:

determining a next dimensional extrema for a next dimension of said n-dimensional array, wherein said next dimensional extrema is related to said dimensional extrema; and

repeating said determining a next dimensional extrema for each of said n-dimensions, wherein each of said next dimensional extrema is related to a dimensional extrema from a previously selected dimension.

30. (currently amended) A computer readable memory device carrying a set of instructions which, when executed, perform a method comprising:

determining within each of said processing elements a dimensional extrema for a first dimension of said n-dimensional array, wherein said dimensional extrema is related to one or more local extrema of said processing elements in said first dimension;

determining within each of said processing elements a next dimensional extrema for a next dimension of said n-dimensional array, wherein said next dimensional extrema is related to one or more of said first dimensional extrema; ~~and~~

repeating said determining within each of said processing elements a next dimensional extrema for each of said n-dimensions, wherein each of said next dimensional extrema is related to a dimensional extrema from a previously selected dimension, until a global extrema is found;  
and

saving said global extrema.